of the slave pipelines 56-59. Since the object 284 is not to be displayed within portions 267 and 269, the screen coordinates of the object 284 should be outside of the ranges rendered by pipelines 57 and 59. Thus, slave pipelines 57 and 59 should discard the graphical data without rendering it to frame buffers 67 and 69. Preferably, bounding box techniques and/or other data optimization techniques are employed to discard the graphical data defining the object 284 before the coordinates of this graphical data are translated to screen relative by pipelines 57 and 59 and/or before other significant processing is performed on this data by pipelines 57 and 59.

Since the top half of the object 284 is to be displayed within portion 266, the screen coordinates of the object should be within the range rendered by pipeline 56 (*i.e.*, from screen coordinates (700, 1000) to (1000, 1300)). Thus, slave pipeline 56 should render the graphical data defining the top half of the object 284 to frame buffer 66. However, since the bottom half of the object 284 is not to be displayed within portion 266, the screen coordinates of the bottom half of the object 284 should be outside of the range rendered by the pipeline 56. Thus, the slave pipeline 56 should discard the graphical data defining the bottom half of the object 284 without rendering this data to frame buffer 66. Preferably, bounding box techniques and/or other data optimization techniques are employed to discard the graphical data defining the bottom half of the object 284 before the coordinates of this graphical data are translated to screen relative by pipeline 56 and/or before other significant processing is performed on this data by pipeline 56.

Since the bottom half of the object 284 is to be displayed within portion 268, the screen coordinates of the object should be within the range rendered by pipeline 58 (*i.e.*, from screen coordinates (700, 700) to (1000, 1000)). Thus, slave pipeline 58 should render the graphical data defining the bottom half of the object 284 to frame buffer 68. However, since the top half of the object 284 is not to be displayed within portion 268, the screen

coordinates of the top half of the object 284 should be outside of the range rendered by the pipeline 58. Thus, the slave pipeline 58 should discard the graphical data defining the top half of the object 284 without rendering this data to frame buffer 68. Preferably, bounding box techniques and/or other data optimization techniques are employed to discard the graphical data defining the top half of the object 284 before the coordinates of this graphical data are translated to screen relative by pipeline 58 and/or before other significant processing is performed on this data by pipeline 58.

As described hereinbefore, the graphical data stored in frame buffers 65-69 should be composited by compositor 76 and rendered to display device 83. The display device 83 should then update the image displayed by the screen 247 such that the object 284 is displayed within portions 266 and 268, as shown by FIG. 10.

Since each pipeline 55-59 renders only a portion of the graphical data defining each image displayed by display device 83, the total time for rendering the graphical data to display device 83 can be significantly decreased, thereby resulting in increased efficiency for the system 50. Thus, in the optimization mode, the speed at which graphical data is rendered from the client 52 to the display device 83 should be maximized. This increase in efficiency is transparent to the application 17, in that the application 17 does not need to be aware of the configuration of the pipelines 55-59 to operate correctly. Thus, the application 17 does not need to be modified to operate successfully in either conventional system 15 or in the system 50 depicted by FIG. 3.

Super-Sampling Mode

Referring to FIG. 3, the operation and interaction of the client 52, pipelines 55-59, and the compositor 76 will now be described in more detail while each of the pipelines 56-59 is operating in the super-sampling mode. In the super-sampling mode, the graphical data

transmitted from the client 52 is super-sampled to enable anti-aliasing of the image produced by display device 83.

For illustrative purposes assume that the application 17, as described hereinabove for the optimization mode, issues a function call for creating an X window 245 having a 3D image displayed within the region 249 of the X window 245, as shown by FIG. 7. In the super-sampling mode, the pipelines 55-59 perform the same functionality as in the optimization mode except for a few differences, which will be described in more detail hereinbelow. More specifically, the client 52 transmits to the master pipeline 55 a command to render the X window 245 and a command to render a 3D image within portion 249 of the X window 245. The command for rendering the X window 245 should include 2D graphical data defining the X window 245, and the command for rendering the 3D image within the X window 245 should include 3D graphical data defining the 3D image to be displayed within region 249. The master pipeline 55 renders the 2D data defining the X window 245 to frame buffer 65 and transmits the 3D data defining the 3D image to slave pipelines 56-59, as described hereinabove for the optimization mode. The master pipeline 55 also assigns the chroma-key to each pixel that is rendered to frame buffer 65 and that is within portion 249.

The slave controller 261 transmits inputs to the slave pipelines 56-59 indicating the range of screen coordinate values that each slave 56-59 is responsible for rendering, as described hereinabove for the optimization mode. Each slave pipeline 56-59 discards the graphical data outside of the pipeline's responsibility, as previously described for the optimization mode. However, unlike in the optimization mode, the pipelines 56-59 super-sample the graphical data rendered by the pipeline 56-59 to frame buffers 66-69, respectively. In super-sampling the graphical data, the number of pixels used to represent the image defined by the graphical data is increased. Thus, a portion of the image